

# OPTIGAN: TOPOLOGICAL OPTIMISATION IN DESIGN FORM-FINDING WITH CONDITIONAL GANS

XUYOU YANG<sup>1</sup>, DING WEN BAO<sup>2,\*</sup>, XIN YAN<sup>3</sup> and YUCHENG ZHAO<sup>4</sup>

<sup>1</sup>*IDesignLab*

<sup>2,\*</sup>*School of Architecture and Urban Design & Centre for Innovative Structures and Materials, RMIT University*

<sup>3</sup>*Tsinghua University*

<sup>4</sup>*ByteDance*

<sup>1</sup>*xuyou.yang.92@gmail.com, 0000-0002-6294-378X*

<sup>2,\*</sup>*nic.bao@rmit.edu.au, 0000-0003-1395-8747*

<sup>3</sup>*yanxin13@mails.ucas.ac.cn, 0000-0002-5033-3597*

<sup>4</sup>*zhaoyucheng.joe@bytedance.com, 0000-0002-9291-2554*

**Abstract.** With the rapid development of computers and technology in the 20th century, the topological optimisation (TO) method has spread worldwide in various fields. This novel structural optimisation approach has been applied in many disciplines, including architectural form-finding. Especially Bi-directional Evolutionary Structural Optimisation (BESO), which was proposed in the 1990s, is widely used by thousands of engineers and architects worldwide to design innovative and iconic buildings. To integrate topological optimisation with artificial intelligence (AI) algorithms and to leverage its power to improve the diversity and efficiency of the BESO topological optimisation method, this research explores a non-iterative approach to accelerate the topology optimisation process of structures in architectural form-finding via conditional generative adversarial networks (GANs), which is named as OptiGAN. Trained with topological optimisation results generated through Ameba software, OptiGAN is able to predict a wide range of optimised architectural and structural designs under defined conditions.

**Keywords.** BESO (Bi-directional Evolutionary Structural Optimisation); Artificial Intelligence; Deep Learning; Topological Optimisation; Form-Finding; GAN (Generative Adversarial Networks); SDG 12; SDG 9.

## 1. Introduction

Structural optimisation, including topology optimisation, plays a significant role in architectural design. It can increase the performance of structures and the efficiency of material use and thus reduce material waste and carbon impact in the fabrication and construction process. By integrating topology optimisation with artificial intelligence

(AI) for more efficient use of materials in the industry, it seeks to help achieve the United Nations Sustainable Development Goal 12: Ensure sustainable consumption and production patterns (United Nations, 2015).

### 1.1. STRUCTURAL OPTIMISATION

Structural optimisations aim to achieve the best structural performance while meeting the requirement of various constraints. Over the past three decades, high-speed computers and rapid improvements in algorithms have been used to develop better structural optimisation solutions by a number of engineering researchers.

#### 1.1.1. Topology Optimisation

Topology optimisation is one of the most popular optimal structural design methods for discrete structures, such as trusses and frames. It is developed to search for the optimal spatial order and connectivity of the bars. Topology optimisation of continuum structures is to find optimal designs by determining cavities' best locations and geometries in the design domains.

Topology optimisation can be readily used to perform shape optimisation by simply restricting the structural modification to the existing boundaries (Huang & Xie, 2010). In the field of topology optimisation, there are several notable methods based on finite element analysis (FEA) developed, such as the homogenisation method (Bendsøe & Kikuchi, 1988), the solid isotropic material with penalisation (SIMP) method (Bendsøe & Sigmund, 1999), the evolutionary structural optimisation (ESO) (Xie & Steven, 1993), the bi-directional evolutionary structural optimisation (BESO) (Huang & Xie, 2010; Huang et al., 2007) and the level-set method (LSM) (Wang et al., 2003). In this paper, bi-directional evolutionary structural optimisation (BESO) proposed by Huang and Xie (2010) is adopted to develop a new integrated topology optimisation algorithm (Figure 1).

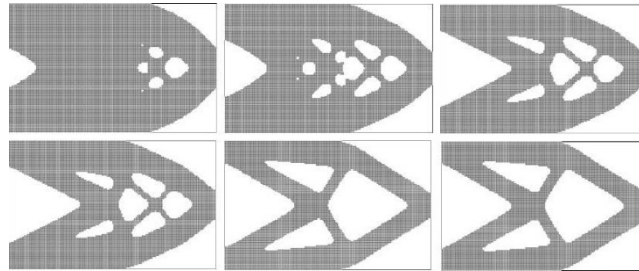


Figure 1 Bi-directional evolutionary structural optimisation (BESO) result

#### 1.1.2. Bi-directional Evolutionary Structural Optimisation (BESO)

Bi-directional evolutionary structural optimisation (BESO) is the emerging technology that is an extension of evolutionary structural optimisation (ESO) developed by Xie and Steven in 1992 (Xie & Steven, 1993). Both ESO and BESO algorithms are based on finite element analysis (FEA) for topology optimisation of continuum structures. BESO algorithm aims to find the solution with the highest structural performance

under certain material limitations by removing or adding material elements step by step (Bao et al., 2020). The ESO method also inspires the Extended ESO method, widely used in architecture design projects, such as the Akutagawa River Side Project in Japan by Ohmori and Qatar National Convention Centre by Arata Isozaki, to generate an optimised model with not only high structural performance but also some different characteristics to meet more functional requirements or aesthetic preferences. In the past few years, Mike Xie and his team have modified many detailed control strategies for topology optimisation in architectural design and development during the process (Yan et al., 2021).

### 1.1.3. Ameba Software

Because of the benefit of form-finding through topology optimisation and Bi-directional evolutionary structural optimisation (BESO), more and more designers and architects seek to use topology optimisation methods to design buildings and furniture. However, due to the complexity and slow speed to directly use the algorithm for architectural design, a new Rhinoceros plug-in named Ameba, a topology optimisation tool based on the BESO method and FEniCS open-source computing platform (Zhou et al., 2018), has been developed. More and more architects and designers have gained opportunities to use this intelligent method to work with computers interactively to create innovative, efficient, and organic architectural forms using Ameba. In this work, the authors use it as the topology optimisation tool to form the dataset for training generative adversarial networks to assist and investigate the research.

## 1.2. GENERATIVE ADVERSARIAL NETWORK AND ITS APPLICATION IN TOPOLOGY OPTIMISATION

Allowed by the development of deep learning algorithms and fast-growing computational power, artificial neural networks, including generative adversarial networks (GANs), have been increasingly used in architectural and structural explorations such as topology optimisation in the design process.

### 1.2.1. Generative Adversarial Networks

A generative adversarial network (GAN) is a particular artificial neural network that learns from a collection of examples and their probability distribution. It is then able to generate more examples from the estimated probability distribution (Goodfellow et al., 2020). A typical GAN often consists of a generator that defines a prior probability distribution  $P(z)$  based on a vector  $z$  as input and a discriminator which examines whether data  $x$  is real (sampled from the training examples) or fake (sampled from the output of the generator).

GANs can further be extended to conditional models (cGANs) where both the generator and discriminator are conditioned on extra information  $y$  as input (Mirza & Osindero, 2014). Besides examining whether  $x$  is real or fake, the discriminator of a cGAN also evaluates whether it matches the condition  $y$ . For example, when using cGANs to solve topology optimisation problems,  $x$  can be the expected optimisation results, given the corresponding boundary and load conditions of  $y$ .

### 1.2.2. Topology Optimisation via Deep Learning

In recent years, there has already been some research into solving topology optimisation problems through artificial neural networks, especially GANs. For example, the TopologyGAN (Nie et al., 2021) is developed on a cGAN, whose generator combines a U-Net (Ronneberger et al., 2015) and ResNet (He et al., 2016). It takes displacement, load boundary conditions and target volume fraction augmented with dense initial fields computed over the unoptimised domain as the input to the model. By doing so, this method vastly improves the accuracy of predicting topology optimisation results compared to some baseline models. Another research by Yu et al. (2019) transforms the boundary conditions into multi-channel images as input to a convolutional-neural-network-based encoder to generate low-resolution topology optimisation results. It inputs the low-resolution results into a GAN to produce final results in high resolution. Differently, the proposed method utilises cGANs directly. It requires very brief input to keep the models easy to operate and thus broaden the potential range of users with or without professional structural knowledge.

### 1.3. PROPOSED METHOD

This research suggests an approach to accelerate the topology optimisation process of structures in architectural form-finding by replacing iterative calculation procedures with an end-to-end algorithm via conditional generative adversarial networks. This method is named OptiGAN by the authors as the ultimate goal is to generate topology optimisation results efficiently and accurately. Trained with a small number of topological optimisation results generated with Ameba software, the proposed method is able to predict a wide range of optimised two-dimensional structural forms under defined conditions.

## 2. Methods

To achieve the research target, a coarse-to-fine network of cGANs is developed and trained with a dataset collected by the authors.

### 2.1. DATA COLLECTION AND PRE-PROCESSING

To train OptiGAN, an original dataset of topology optimisation results is collected with Ameba software. In detail, the material is kept as steel during the data generating process, and the volume fraction is set to 0.5 consistently. The parameters that can vary from case to case are design domain, fixing edge and load conditions, which are the very parameters to use as input parameters of OptiGAN. In practice, in the Ameba script, the fixing edge of a square design domain is always set to the left edge as fixing at other edges are seen as the same as rotating left-edge-fixing conditions in the later data augmentation process when training the cGAN models. In the first stage of the research, a total number of 1385 optimisation results are included in the dataset.

During the pre-processing, the input parameters are translated into a three-channel input image at  $256 * 256$  pixels in size. By doing so, the model is provided with two-dimensional spatial clues. Thus, the difficulty for training the model to predict two-dimensional results can be reduced compared to using merely numerical inputs directly without spatial suggestions. Specifically, as Figure 2 demonstrates, the design domain and fixing edge are expressed in all three channels by assigning the value of 0 to the corresponding pixels representing the geometries. Load locations and load unit vectors projected onto the X and Y axis are documented in the first and second channel respectively in values remapped into a range between 0 and 255. Values of the other pixels are assigned 255 in all three channels by default.

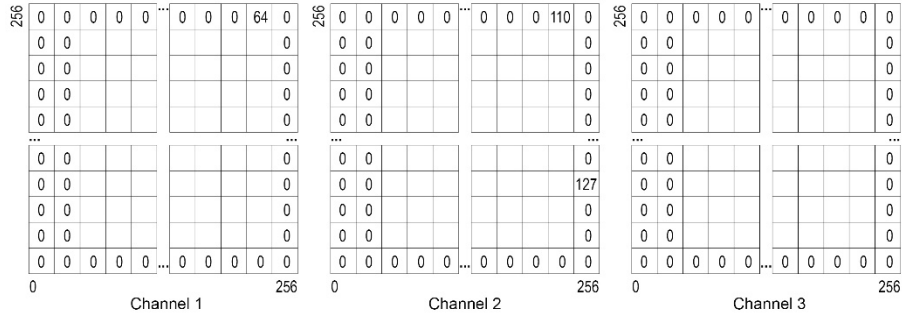


Figure 2 Translated input image of  $1000*1000$  mm design domain, fixing at the left edge, load one at  $(900, 1000)$  location in  $45^\circ$  angle, load two at  $(1000, 700)$  location in  $180^\circ$  angle. The values of the rest pixels are all 255.

## 2.2. OPTIGAN ARCHITECTURE

Unlike the previous researches mentioned in section 1.2.2, there are no dense initial fields, low-resolution results, or any other inputs than design domain, fixing edge and load conditions for the OptiGAN generator. Keeping the inputs simple can make it potentially as easy as adjusting a few number sliders for the OptiGAN users to operate. At the same time, it dramatically increases the difficulty for the generator to speculate the results by providing less input information. To respond to the conflict, OptiGAN adopts a coarse-to-fine network architecture.

Specifically, as Figure 3 demonstrates, there are two generators and discriminators in the network. The initial input of conditions in the form of translated input images is first fed to the coarse generator, which then predicts a rough output examined by the coarse discriminator. Then the rough output together with the initial conditions are input to the fine generator, which outputs the final results. Although compared to conventional cGANs, the input of OptiGAN consists only the conditions  $y$  alone without vector  $z$ , the networks can also learn from only the conditions. This choice of input is also being suggested in Pix2Pix (Isola et al., 2017), one of the most successful image-to-image translation models.

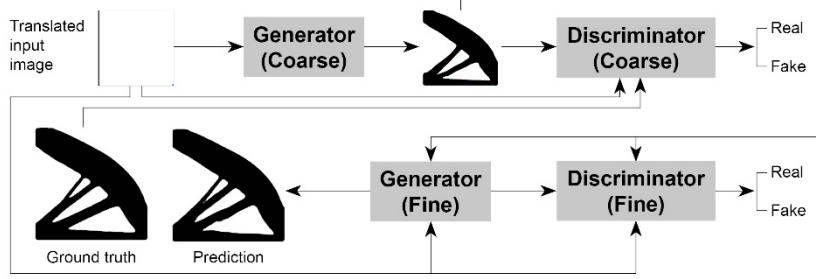


Figure 3 The architecture of OptiGAN

### 2.2.1. Generators and Discriminators

Both the coarse and fine generators used in OptiGAN are U-Net (Ronneberger et al., 2015), which has a mirrored encoder-decoder network architecture with skip connections between symmetrical layers. It can work effectively with very few training data. Meanwhile, both discriminators are PatchGAN (Isola et al., 2017), which focuses on  $156 * 156$  patches as through testing, such patch size provides the best outcome in the tasks.

### 2.2.2. Objective

The objective that OptiGAN tries to optimise can be expressed as equation (1). It consists of two parts, the cGAN loss (equation (2)) and L1 loss (equation (3)). Besides generating images that look real, as the other goal is to eventually achieve results as close to the BESO optimisation outcomes as possible, L1 loss is added to the total loss with a considerable weight of  $\lambda$  to force the output to be close to the ground truth. L1 loss is chosen out of L2 loss because it encourages less blurring effect of images compared to L2 loss. In the experiments, the weight is set to 125 ( $\lambda = 125$ ) to emphasise the importance of L1 loss in this particular task.

$$G = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \quad (1)$$

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_x[\log(1 - D(x, G(x)))] \quad (2)$$

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y}[\|y - G(x)\|_1] \quad (3)$$

## 2.3. TRAINING OPTIGAN

OptiGAN is trained in two steps: the coarse generator and discriminator are trained first, after which the fine generator and discriminator are trained. Both parts of the network are trained for 200 epochs to achieve the demonstrated results. Following the conventions, instead of minimising  $\log(1 - D(x, G(x)))$ , the trainings maximise  $\log(D(x, G(x)))$  to avoid saturating  $\log(1 - D(x, G(x)))$  in the early training stage. The initial learning rate is 0.0005 decaying to 0 in the last 100 epochs and the Adam optimiser is used. Figure 4 shows the history of the cGAN loss and L1 loss during the

two training procedures.

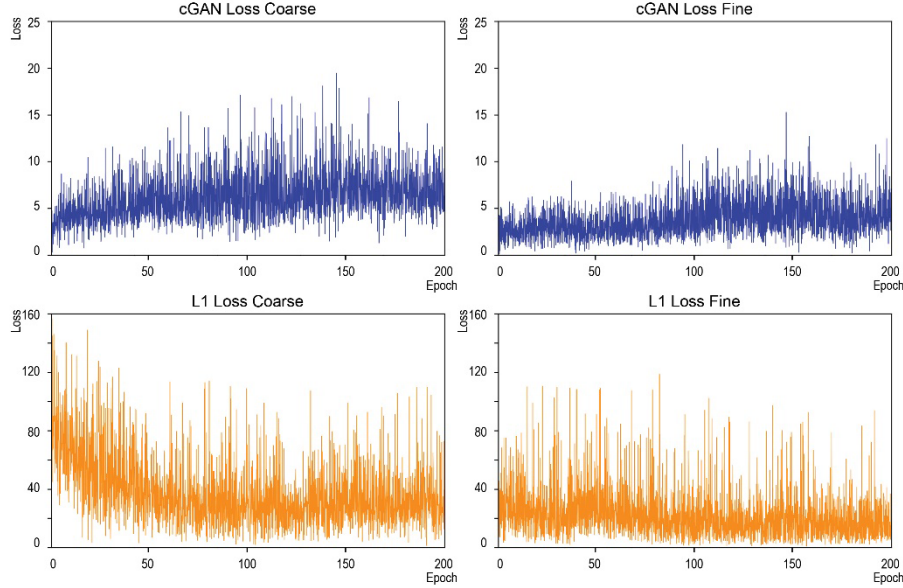


Figure 4 The cGAN loss and L1 loss during the coarse training and fine training

The coarse generator and fine generator take different types of input and are paired with different discriminators as introduced in section 2.1.1, so it is not considered practical to compare the absolute value of the corresponding cGAN losses of the coarse and fine models. Even though, the cGAN loss improves from the beginning to the last epoch in both cases, as shown in Figure 4. In contrast, for both generators, the L1 loss is calculated according to the same ground truth, and it can be discovered that the fine generator further decreases the L1 loss based on the coarse generator, which indicates that the fine generator is able to further improve the accuracy of predicted topology optimisation results by OptiGAN.

### 3. Results

During the training process, the L1 loss of OptiGAN reduced from over 80 to less than 20. More importantly and precisely, the pixel-wise accuracy is used to evaluate the performance of the models. It is equal to the percentage of accurate pixels in a prediction out of the total pixels of that image. Tested with 150 randomly selected pieces of data different from the training set, the average pixel-wise accuracy of OptiGAN is able to achieve 83.15%. Figure 5 demonstrates some of the testing results in different load conditions with pixel-wise differences between the predictions and ground truth visualised for each case.

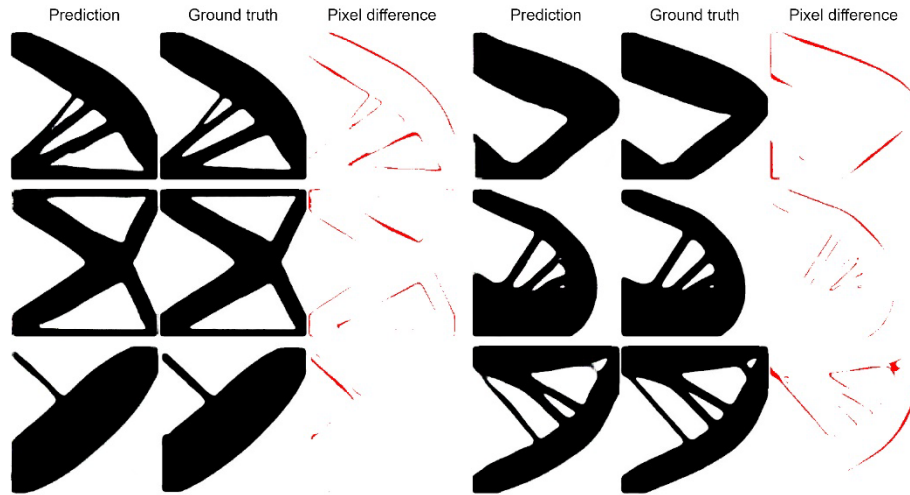


Figure 5 Some prediction results of OptiGAN with ground truth and pixel-wise difference

#### 4. Discussion

OptiGAN demonstrates the ability of a novel approach and its application in architectural and structural form-finding. It is the extension of the SwarmBESO (multi-agent-based topology optimisation) method proposed by Bao & Yan in 2020 (Bao et al., 2021) to improve the diversity of the topological optimisation generative method. It has the potential to significantly help architects and engineers save material and produce more efficient structural layouts and building envelopes. It is valuable to integrate two intelligent computational design methods, deep learning and topology optimisation, for designers in the conceptual design phases.

However, the research is in a rudimentary phase and is temporarily constrained in a range of two-dimensional solutions. Although set as an input variable, the design domain in the current dataset includes only square geometries, despite that it can perform well in this geometric range, as demonstrated in the testing results. To truly diversify the spectrum of results and further increase the accuracy, it is very crucial that OptiGAN must be trained with much more data of various design domains and load conditions. Future line of research also includes further equipping the model with the ability to solve three-dimensional topology optimisation problems.

#### 5. Conclusion

This research develops OptiGAN, a non-iterative method to accelerate the topology optimisation process of structures in architectural form-finding via conditional generative adversarial networks with high accuracy. It demonstrates the process of integrating topology optimisation and generative adversarial networks to establish an artificial intelligence (AI) based structural optimisation technique. This new methodology holds great potential for practical application in architecture and

engineering fields. It increases the diversity of outcome of the topology optimisation generative design such as the application of shell (Figure 6).

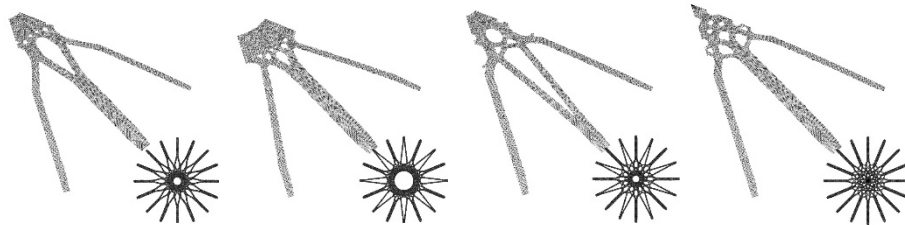


Figure 6 Diverse BESO results of shell optimisation

### Acknowledgements

We thank Nanjing Ameba Engineering Structure Optimization Research Institute for providing educational version of Ameba software to support our research. (For Ameba, see Ameba.xieym.com)

### References

- Bao, D. W., Yan, X., Snooks, R., & Xie, Y. M. (2020). Bioinspired Generative Architectural Design Form-Finding and Advanced Robotic Fabrication Based on Structural Performance. In *Architectural Intelligence* (pp. 147-170), Springer.
- Bao, D. W., Yan, X., Snooks, R., & Xie, Y. M. (2021). SwarmBESO: Multi-agent and evolutionary computational design based on the principles of structural performance. In *26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia: PROJECTIONS CAADRIA 2021*, Hong Kong (pp. 241-250).
- Bendsøe, M. P., & Kikuchi, N. (1988). We are generating optimal topologies in structural design using a homogenisation method. *Computer methods in applied mechanics engineering*, 71(2), 197-224. doi:[https://doi.org/10.1016/0045-7825\(88\)90086-2](https://doi.org/10.1016/0045-7825(88)90086-2)
- Bendsøe, M. P., & Sigmund, O. (1999). Material interpolation schemes in topology optimisation. *Archive of applied mechanics*, 69(9), 635-654. doi:<https://doi.org/10.1007/s004190050248>
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139-144. doi:10.1145/3422622
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, (pp. 770-778).
- Huang, X., & Xie, M. (2010). *Evolutionary topology optimisation of continuum structures: methods and applications*: John Wiley & Sons.
- Huang, X., Xie, Y. M., & Burry, M. C. (2007). Advantages of bi-directional evolutionary structural optimisation (BESO) over evolutionary structural optimisation (ESO). *Advances in Structural Engineering*, 10(6), 727-737. doi:<https://doi.org/10.1260/136943307783571436>
- Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1125-1134).

- Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv: 1411.1784*.
- Nie, Z., Lin, T., Jiang, H., & Kara, L. B. (2021). Topologygan: Topology optimisation using generative adversarial networks based on physical fields over the initial domain. *Journal of Mechanical Design*, 143(3), 031715.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, (pp. 234-241).
- United Nations. (2015, September 25). *Transforming our world: The 2030 agenda for sustainable development*. Retrieved January 20, 2022, from: [https://https://www.un.org/ga/search/view\\_doc.asp?symbol=A/RES/70/1&Lang=E](https://https://www.un.org/ga/search/view_doc.asp?symbol=A/RES/70/1&Lang=E).
- Wang, M. Y., Wang, X., & Guo, D. (2003). A level set method for structural topology optimisation. *Computer methods in applied mechanics engineering*, 192(1-2), 227-246. doi:[https://doi.org/10.1016/S0045-7825\(02\)00559-5](https://doi.org/10.1016/S0045-7825(02)00559-5)
- Xie, Y. M., & Steven, G. P. (1993). A simple evolutionary procedure for structural optimisation. *Computers & structures*, 49(5), 885-896. doi:[https://doi.org/10.1016/0045-7949\(93\)90035-C](https://doi.org/10.1016/0045-7949(93)90035-C)
- Yan, X., Bao, D., Zhou, Y., Xie, Y., & Cui, T. (2021). Detail control strategies for topology optimisation in architectural design and development. *Frontiers of Architectural Research*, 10, 1-17. doi:<https://doi.org/10.1016/j.foar.2021.11.001>
- Yu, Y., Hur, T., Jung, J., & Jang, I. G. (2019). Deep learning for determining a near-optimal topological design without any iteration. *Structural Multidisciplinary Optimization*, 59(3), 787-799. doi:<https://doi.org/10.1007/s00158-018-2101-5>
- Zhou, Q., Shen, W., Wang, J., Zhou, Y. Y., & Xie, Y. M. (2018). Ameba: A new topology optimisation tool for architectural design. In *International Association for Shell and Spatial Structures (IASS)*, (pp. 1-8).